SPECIFICATION AMENDMENTS:

Page 1, lines 4-6

The present invention relates generally to JPEG image compression, and more specifically to a method and apparatus for optimizing a JPEG Part 1 image using regionally variable compression levels.

Page 1, line 9 to Page 2, line 5

Compression is a useful method for reducing bandwidth consumption and download times of images sent over data networks. A variety of algorithms and techniques exist for compressing images. JPEG, a popular compression standard that is particularly good at compressing photo-realistic images, is in common use on the Internet. This standard, ~~defined~~ described in ``JPEG Still Image Data Compression Standard'', by W.B. Pennebaker and J. L. Mitchell, Chapman & Hall, 1992, is based on a frequency domain transform of blocks of image coefficients. As seen in Figure **1**, JPEG calls for subdividing an image frame **12** into 8x8 pixel blocks **11** and at box **16** transforming the array of pixel values in each block **11** with a discrete cosine transform (DCT) so as to generate 64 DCT coefficients corresponding to each pixel block **11**. The coefficients for each block **11** are quantized in quantization block **20** using a 64 element quantization table **24**. Each element of table **24** is an integer value from 1 to 255, which specifies the step size of the quantizer for the corresponding DCT coefficients. The quantized coefficients for each block are entropy encoded in entropy coding box **28**, which

performs a lossless compression. The entropy encoder **28** is coupled to the output of the quantizer **20** from which the former receives quantized image data. Standard JPEG entropy coding uses either Huffman coding or arithmetic coding using either predefined tables or tables that are computed for a specific image.

Page 2, lines 18-23

More specifically, the discrete cosine transform block uses the forward discrete cosine function (DCT) to transform the image pixel intensity ~~A(x,y)~~X[x,y] to DCT coefficients ~~Y~~$_{mn}$Y[m,n] as follows:

$$\text{~~Y~~}_{mn}Y[m,n] = 1/4\ C(m)C(n)\left[\sum_{x=0}^{7}\sum_{y=0}^{7} \text{~~A(x,y)~~}X[x,y]\cos\frac{(2x+1)m\pi}{16}\cos\frac{(2y+1)n\pi}{16}\right]$$

where C(m) and C(n)=1/$\sqrt{2}$ for m,n=0, and C(m) and C(n)=1 ~~otherwise~~.

Page 3, line 1 - Page 4, line 4

The next step is to quantize the DCT coefficients using a quantization matrix, which is an 8 x 8 matrix of step sizes with one element for each DCT coefficient. A tradeoff exists between the level of image distortion and the amount of compression, which results from the quantization. A large quantization step produces large image distortion, but increases the amount of compression. A small quantization step produces

lower image distortion, but results in a decrease in the amount of compression. JPEG typically uses a much higher step size for the coefficients ~~which~~ correspond~~ing~~ing to high spatial frequency in the image, with little noticeable deterioration in the image quality because of the human visual system's natural high frequency roll-off. The quantization is actually performed by dividing the DCT coefficient ~~Y~~$_{mn}$Y[m,n] by the corresponding quantization table entry ~~Q~~$_{mn}$-Q[m,n] and the result rounded off to the nearest integer according to the following:

$$\text{~~T~~}_{mn}T[m,n] = \text{round}(\text{~~Y~~}_{mn}Y[m,n]/\text{~~Q~~}_{mn}Q[m,n])$$

to give a quantized coefficient ~~T~~$_{mn}$T[m,n]. This type of quantizer is sometimes referred to as a midtread quantizer. An approximate reconstruction of ~~Y~~$_{mn}$-Y[m,n] is effected in the decoder by entrywise ~~multiplying~~ multiplication of ~~T~~$_{mn}$T[m,n] by ~~Q~~$_{mn}$-Q[m,n] to obtain a reconstructed Z[m,n]:

$$Z[m,n] = Q[m,n]\ T[m,n]\text{~~÷~~}$$

The difference between ~~Y~~$_{mn}$-Y[m,n] and ~~Y~~$_{mn}$-Z[m,n] represents lost image information causing distortion to be introduced. The amount of this lost information ~~depends on~~ is bounded by the magnitude of ~~Q~~$_{mn}$Q[m,n].

Page 4, lines 6 to line 21

In the case of an image with multiple color channels, the aforementioned steps are applied in a similar fashion to each channel independently. ~~In some cases~~ In general practice, ~~some of~~ the color channels ~~may be~~ are sub-sampled to achieve greater compression, without significantly altering the quality of the image reconstruction.

The quantization step is of particular interest since this is where information is discarded from the image. Ideally, one would like to discard as much information as possible, thereby reducing the stored image size, while at the same time maintaining or increasing the image fidelity. Within the standard there is no prescribed method of quantizing the image, but there is nonetheless a popular method used in the software of the Independent JPEG Group (ISO/IEC JTC1 SC29 Working Group 1), and employed extensively by the general community.

Page 7, line 6 to Page 8, line 9

Accordingly, it is an object of the invention to provide a method for quantizing a JPEG image, which offers many of the benefits of variable quantization and is computationally efficient, while conforming to the widely used JPEG Part 1 standard.

SUMMARY OF THE INVENTION

According to the present invention there is provided a method, which is directed towards regionally variable levels of compression. The method is directed to JPEG compression of an image frame divided up into non-overlapping 8 x 8 pixel blocks ~~B_{ij}~~ X_i ~~where I, j are integers covering all of the blocks in the image frame~~. The method includes forming a discrete cosine transform (DCT) of each block ~~B_{ij}~~ X_i of the image frame to produce a matrix of blocks of transform coefficients ~~D_{ij}~~ Y_i. Next a visual importance, I_i, is calculated for each block of the image, —based upon assigning zeros for flat features and values approaching unity for sharply varying features. A global quantization matrix Q is then formed such that the magnitude of each quantization matrix coefficient ~~Q_{ij}~~ Q[m,n] is inversely proportional to ~~a visual importance I_{ij}~~ the aggregate visual importance of ~~a~~ the corresponding DCT basis vector to the image. The local visual importance is used during the quantization stage, where for regions of lower detail, more data is discarded, resulting in more aggressive compression. In the quantization stage the transform coefficients are pseudo-quantized by ~~dividing them by a factor~~ the quantization matrix S_i Q, where S_i is a linear scaling factor, to create a JPEG Part 1 image file. This algorithm is unique in that it allows for the effect of variable-quantization to be achieved while still producing a file which conforms to the JPEG Part 1 standard.

Page 8, line 11 to Page 12, line 14

The visual importance, I_i may be determined by discrete edge detection and summation of transform coefficients. This determination of I_i may include creating a 24 x 24 matrix of image pixels of DCT coefficients centered on a block ~~BX_{ij}~~ ~~where I and j = 1,2,...,8~~. The center 10 x 10 matrix of the 24 x 24 matrix may be convolved with an ~~edge~~ edge-tracing kernel. The ~~matrix~~ values of the convolved matrix may be summed, and the summed value normalized to produce a visual importance, I_{ij}.

The global quantization matrix, Q, may be formed by calculating an 8 x 8 matrix A by calculating matrix elements ~~A_{mn}~~A[m,n] according to the formula:

~~$A_{mn} = I_{ij}(B_{ij})_{mn}$.~~

$$A[m,n] = \sum_{\text{all } i} I_i \, Y_i[m,n]$$

~~Elements Qmn~~The elements of an intermediate matrix ~~Q~~ B may then be calculated according to the formula:

~~$Q_{mn} = \max(A_{mn})/A_{mn}$~~

$$B[m,n] = A_{max} / A[m,n]$$

where A_{max} = max{all entries of A}. A scaling factor s is calculated that minimizes the quantity $|s\,B - Q_{std}|$, where Q_{std} is a chosen standard quantization matrix. The quantization matrix is then calculated as Q = s B.

The linear scaling factors S_i may be set equal to $I_i(S_{max}-S_{min}) + S_{min}$, where S_{max} and S_{min} are user selected.

~~Quantizing the blocks of DCT coefficients D_{ij} to produce quantized DCT coefficients T_{ijm}, where m and n refer to row and column, respectively, in each of the blocks may be accomplished by applying the formula~~

~~$T_{ijmn} = \text{round}(D_{ijmn}/(S_{min}*Q_{mn}))$, where round denotes rounding to the nearest integer,~~

~~and if $T_{ijmn} \neq 0$~~

~~calculate round $(D_{ijmn}/(S_{ij}*Q_{mn}))$ and if equal to zero then set $T_{ijmn} = 0$, otherwise if~~

~~$(\text{abs}(D_{ijmn}) - (2\hat{}(\text{ceil}(\text{lg}(\text{abs}(D_{ijmn}) + 1)) - 1) - 1)))$~~

~~$\leq \text{abs}(D_{ijmn}Q_{mn} - S_{ij}*\text{round}(D_{ijmn}/(S_{ij}*Q_{mn})))$~~

~~then~~

~~$T_{ijmn} = \text{sign}(D_{ijmn})*(2\hat{}(\text{ceil}(\text{lg}(\text{abs}(D_{ijmn}) + 1)) - 1)$.~~

Variable quantization may be approximated by first uniformly quantizing a DCT coefficient block Y_i by the global quantization matrix $Q_{min} = S_{min} Q$ according to the standard formula:

$$T_i[m,n] = \text{round}(Y_i[m,n]/Q_{min}[m,n])$$

Let $P(x)$ be a function that returns the nearest integer of the form $2^k - 1$ that is smaller than the positive integer x:

$$P(x) = 2^{\text{floor}(\lg(x))} - 1$$

Let $Evq_i[m,n]$ be the error that would be introduced to the coefficient $Y[m,n]$ by uniform quantization with a local quantization matrix $S_i Q$, and let $Ernd_i[m,n]$ be the error introduced by rounding down the coefficient $T_i[m,n]$ to the nearest smaller integer of the form $2^k - 1$:

$$Evq_i[m,n] = |S_i Q[m,n] \text{round}(Y_i[m,n]/S_i[m,n]Q[m,n])) - Y_i[m,n].$$

$$Ernd_i[m,n] = |Q_{min}[m,n] P(\text{abs}(T_i[m,n])) - \text{abs}(Y_i[m,n])|$$

Variable quantization may be approximated by modifying each uniformaly quantized coefficient $T_i[m,n]$ as follows:

1. If $\text{round}(Y_i[m,n]/(S_i Q)) = 0$, then $T_i[m,n] = 0$.
2. Otherwise, if $Ernd_i[m,n] <= Evq_i[m,n]$ then $T_i[m,n] = \text{sign}(T_i[m,n]) P(\text{abs}(T_i[m,n]))$.

Step 1 serves to erase any coefficients that would have been erased had they truly been quantized by a local quantization matrix $S_i Q$. Step 2 decreases values in order to guarantee a smaller Huffman representation if the error introduced in doing

so is less than or equal to the error that would have been introduced by variable quantization.

According to another aspect of the invention there is provided a method of JPEG compression of a colour image represented by channels Y for greyscale data, and U and V each for colour, which comprises shrinking the colour channels U and V by an integer fraction of their size, forming a discrete cosine transform (DCT) ~~D_{ij}~~ Y_i for each block ~~B_{ij}~~ X_i of each of channels Y, U and V and calculating a visual importance, I_i, for each Y channel block of each image and setting I_i = max{ I_i values for corresponding Y channel blocks} for blocks in the U and V channels. A global quantization matrix Q is formed for the Y channel block and one for channels U and V combined such that a magnitude of each quantization matrix coefficient ~~Q_{ij}~~ Q[m,n] is inversely proportional to an aggregate visual importance ~~I_{ij} to~~ in the image of ~~a~~ the corresponding DCT basis vector. Next the transform coefficients for each of the Y, U and V channels are variable-quantized by dividing them by a factor the matrix ~~S_{ij}~~ S_{min} ~~Q^1~~ and local parameter S_i, where ~~S_{ij}~~ S_i is a linear scaling factor ~~for each of chanels Y, U and V and,~~ ~~Q^1~~ is the global quantization table for the associated channel being quantized, and S_{min} is a user determined scaling factor. Finally, the quantized coefficients ~~T_{ijmn}~~ Y{m,n] the global uniform quantization table $Q_{min} = S_{min} Q$ are entropy encoded, ~~where S_{min} is a user selected minimum scaling factor for each of channels Y, U, and V,~~ to create a JPEG Part 1 image file for each of channels Y, U and V.

Preferably, the ~~shrinking~~ subsampling factor is ~~½~~ 2.

In another aspect of the invention there is provided an apparatus for JPEG compression of an image frame divided up into a plurality of non-overlapping, tiled 8 x 8 pixel blocks ~~B_{ij}~~ X_i ~~where I, j are integers covering all of the blocks in the image frame~~. The apparatus includes a discrete cosine transformer (DCT) operative to form the discrete cosine transform of each block ~~B_{ij}~~ X_i of the image frame to produce ~~a matrix of~~ blocks of transform coefficients ~~D_{ij}~~ Y_i, a visual importance calculator operative to calculate the visual importance, ~~I_{ij}~~ I_i, for each block of the image, based upon assigning zeros for flat features and values approaching unity for sharply varying features and a global quantization matrix calculator operative to calculate the global quantization matrix, Q, by one of

(i) selecting a standard JPEG quantization table ~~or~~ and

(ii) selecting a quantization table such that the magnitude of each quantization matrix coefficient ~~Q_{ij}~~ $Q[m,n]$ is inversely proportional to the visual importance in the image of the corresponding DCT basis vector.

A linear scaling factor calculator determines a linear scaling factor, ~~S_{ij}~~ S_i, ~~defining bounds over which the image is to be variably quantized~~ based on user established values of S_{max} and S_{min}, and the visual importance value I_i. A pseudo-variable-

quantizer ~~is operative to divide the transform coefficients, D_{ijmn}, by a value equivalent to dividing them by a factor S_{min} is a user selected minimum scaling factor, and an entropy encoder encodes the quantized coefficients T_{ijmn} and $Q * S_{min}$ to create a JPEG image file.~~ emulates the effects of quantizing a block Y_i by the local quantization matrix S_i Q but produces coefficients T_i actually uniformly quantized by the global quantization matrix S_{min} Q. An entropy coder encodes the quantized coefficients T_i and global quantization matrix S_{min} Q to create a JPEG Part 1 image file.

Page 12, line 21 to Page 20, line 16

For each 8x8 block ~~B_{ij}~~ X_i in the image frame, a visual image importance ~~I_{ij}~~ I_i is calculated at step **44**. Note that the actual measure of visual importance is not important to the outline of the algorithm. The ~~I_{ij}~~ I_i values exhaustively cover the range [0,1], and are a measure of how aggressively the block can be quantized. A value of ~~I_{ij}~~ I_i = 0 indicates that the visual appearance of the block is rather insensitive to the level of quantization, and a value of ~~I_{ij}~~ I_i = 1 indicates that the visual appearance of the block is very sensitive to the level of quantization.

One method of selecting the visual importance ~~I_{ij}~~ I_i is based on a discrete edge-detection and summation technique. Consider a 24 x 24 window ~~W_{ij}~~ Wi on the image defined by the nine image blocks ~~$B_{i-1,j-1}$, $B_{i,j-1}$, $B_{i+1,j-1}$, $B_{i,j}$ $B_{i+1,j}$, $B_{i-1,j+1}$, $B_{i,j+1}$, $B_{i+1,j+1}$.~~

~~This window is~~ centered around the block ~~B_{ij}~~X_i. ~~The nine blocks are shown graphically in the following diagram:~~

~~$B_{i-1,j-1}$~~	~~$B_{i,j-1}$~~	~~$B_{i+1,j-1}$~~
~~$B_{i-1,j}$~~	~~$B_{i,j}$~~	~~$B_{i+1,j}$~~
~~$B_{i-1,j+1}$~~	~~$B_{i,j+1}$~~	~~$B_{i+1,j+1}$~~

From this 24 x 24 window, the matrix V_i of the center ~~a~~ 10 x 10 ~~window V_{ij}, centered about B_{ij}, is~~ pixels are then convolved with a standard Laplacian edge detection kernel G, to give ~~H_{ij}~~H_i. The edge detection kernel employed is,

$$G = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

and the convolution is given by,

$$H_i[m,n] = \sum_{x,y} V_i[m-x,n-y]\ G[x,y]$$

~~$H_{ij} = \sum V_{i-m,j-m} G_{ij}$~~

This technique is essentially the discrete equivalent of taking the second derivative of the image in both dimensions. The output of the convolution ~~H_{ij}~~ $\underline{H_i}$ is scaled to cover an 8-bit range between 0 and 255, the same range taken by the actual pixels in the image. The convolved values are then summed, and the sum is divided by 100*255 to scale the sum to the range 0 to 1. This scaled sum is denoted as ~~K_{ij}~~ $\underline{K_i}$. This sum is then renormalized using the following function:

~~$I_{ij} = \frac{K_{ij}(100+C)}{100K_{ij}+C}$~~ $\underline{I_i = K_i\ (100 + C)\ /\ (100\ K_i + C)}$

where C is equal to 14. This function is determined statistically, and remaps the K_i values such that they lie on a normal distribution.

The above procedure is used to calculate ~~I_{ij}~~ $\underline{I_i}$ for each block in the image. The end result is a value for each ~~I_{ij}~~ $\underline{I_i}$ which is bounded on the ~~region~~ interval ~~(0,1)~~ [0,1], takes values of 0 for flat blocks, and values approaching 1 for blocks that have lots of sharp, short features (in other words have large second derivatives).

The quantization matrix Q is determined at step **46**. In one approach, Q is simply set equal to the standard JPEG

quantization table, which is in general use~~,~~ by the community. An example of a suitable such matrix is the following:

```
 6,  11,  10,  16,  24,  40,  51,  61,
12,  12,  14,  19,  26,  58,  60,  55,
14,  13,  16,  24,  40,  57,  69,  56,
14,  17,  22,  29,  51,  87,  80,  62,
18,  22,  37,  56,  68, 109, 103,  77,
24,  35,  55,  64,  81, 104, 113,  92,
49,  64,  78,  87, 103, 121, 120, 101,
72,  92,  95,  98, 112, 100, 103,  99
```

In another approach, an image-specific quantization matrix is generated, where the magnitude of each quantization table coefficient is inversely proportional to the importance in the image of the corresponding basis vector.

One approach to generating an image-specific quantization matrix Q defines an 8x8 array such that each value ~~A_{mn}~~ A[m,n] is equal to the sum of the corresponding coefficients (m,n) in each block ~~B_{ij}~~ Y_i, weighted by the importance value ~~I_{ij}~~ I_i:

~~$A_{mn} = \sum_{\forall(i,j)} I_{ij} (B_{ij})_{mn}$~~ $A[m,n] = \sum_{\text{all } i} I_i \, Y_i[m,n]$

After this summation, the matrix A holds relative counts of importance for each basis vector in the DCT transform. This matrix is simply inverted and scaled entry-wise such that ~~$\forall_{mn} = \max(A_{mn})/A_{mn}$~~ B[m,n] = max{entries of A} / A[m,n]. In the cases where ~~A_{mn}~~ A[m,n] is zero, ~~A_{mn}~~ B[m,n] is set to 255, which is the largest allowable value for an 8 bit number. The values in B[m,n]

~~A_{mn}~~ are then scaled by a factor s such that the squared error between sB ~~A_{mn}~~ and ~~the~~ a standard ~~JPEG~~ quantization matrix is minimized. The quantization matrix Q is then set equal to this scaled matrix sB. Note that this process is only performed on the AC coefficients, in other words for all values of (m,n) except (0,0). For the (0,0) entry, Q_{00} is simply initialized to the corresponding value in the standard ~~JPEG~~ quantization table.

Each block ~~B_{ij}~~ X_i is DCT transformed at step **48** according to the JPEG standard to produce DCT coefficients ~~D_{ij}~~ Y_i.

For each block ~~B_{ij}~~ X_i in the image, a value ~~S_{ij}~~ S_i is calculated at step **50** where ~~S_{ij}~~ S_i = ~~I_{ij}~~ I_i * $(S_{max} - S_{min}) + S_{min}$. The parameters S_{max} and S_{min} are user specified and in effect define the quality bounds over which the image will be variably quantized. This method is preferably used to remove redundant data from an existing compressed JPEG by letting S_{min} be equal to the actual scaling value used in compressing the image originally, and using a user-defined value for S_{max}.

Each block ~~B_{ij}~~ X_i in the image is "pseudo-quantized" at step **56** with the quantization table ~~Q_{mn}~~ ~~S_{ij}~~ S_i Q. This pseudo-quantization in effect emulates variable quantization while actually quantizing with the global quantization matrix S_{min} Q. If one lets ~~D_{ij}~~ Y_i be the original unquantized DCT transformed image block, and ~~T_{ij}~~ T_i the quantized DCT transformed block at step **54**,

then the algorithm for the pseudo-quantization can be described as given next.

The algorithm has three distinct quantization steps. In the first step, the coefficients in the block ~~B_{ij}~~ Y_i are quantized using the standard JPEG quantization function with S_{min} as the scaling value:

for each block ~~D_{ij}~~ Y_i do

for each coefficient ~~D_{ij}~~ $Y_i[m,n]$ in block ~~D_{ij}~~ Y_i do

~~$T_{ijmn} = round(D_{ijmn}/(Q_{mn}*S_{min}))$~~ $T_i[m,n] = round(Y_i[m,n]/(S_{min}\ Q[m,n])$

where round denotes rounding to the nearest integer.

In the next step, if any coefficient ~~T_{ijmn}~~ $T_i[m,n]$ is > 0, then

if ~~$round(D_{ijmn}/(Q_{mn}*S_{ij}))$~~ $round(Y_i[m,n]/(S_i\ Q[m,n])) = 0$ then ~~T_{ijmn}~~ $T_i[m,n] = 0$

In the third and final step, if ~~T_{ij}~~ $Ti[m,n]$ is still greater than zero, and if the coefficient can be rounded down by one logarithm base-2 and not exceed the rounding error introduced by the quantization with the local quantization table, then it is so rounded down:

if $Ernd_i[m,n] <= Evq_i[m,n]$ then

$T_i[m,n] = sign(T_i[m,n])\ P(abs(T_i[m,n]))$

where $Ernd_i$, Evq_i and P are as defined earlier.

~~if $(abs(D_{ijmn}) - 2\hat{}(ceil(lg(abs(D_{ijmn})+1))-1)-1))$~~

~~$<= abs(D_{ijmn}-Q_{mn}*S_{ij} * round(D_{ijmn}/(Q_{mn}*S_{ij})))$~~

~~then~~

~~$T_{ij} = sign(D_{ijmn})*(2\hat{}(ceil(lg(abs(D_{ijmn})+1))-1)-1)$~~

~~In the above calculations, $Q_{mn}*round(D_{ijmn}/(S_{ij}*Q_{mn}))$ is the reconstructed coefficient after quantization by the local quantization table, and~~

~~$abs(D_{ijmn}-Q_{mn}S_{ij}*round(D_{ijmn}/(S_{ij}*Q_{mn})))$~~

~~is the absolute error introduced by quantization. Furthermore,~~

~~$ceil(lg(abs(D_{ijmn})+1))$~~

~~is the logarithm base-2 of the magnitude of the coefficient, and,~~

~~$(2\hat{}(ceil(lg(abs(D_{ijmn})+1))-1)-1)$~~

~~is the magnitude of the coefficient rounded down by a logarithm base-2.~~

~~Thus,~~

~~abs(D$_{ijmn}$) - (2^(ceil(log(abs(D$_{ijmn}$)+1)) - 1) - 1)~~

~~is the absolute error introduced by rounding down by one logarithm base 2.~~

The algorithm in its entirety is:

for each block ~~D$_{ij}$~~ Y$_i$ do

{

for each coefficient ~~D$_{ij}$~~ Y$_i$[m,n] in block ~~D$_{ij}$~~ Y$_i$ do

{

~~T$_{ijmn}$ = round(D$_{ijmn}$/(S$_{min}$*Q$_{mn}$))~~ T$_i$[m,n] = round(Y$_i$[m,n]/(S$_{min}$ Q[m,n]))

if ~~T$_{ijmn}$~~ T$_i$[m,n] > 0 then

{

if round(~~D$_{ijmn}$/S$_{ij}$*Q$_{mn}$~~)) roundY$_i$[m,n]/(S$_i$ Q[m,n])) = 0 then ~~T$_{ijmn}$~~ T$_i$[m,n] = 0

else

{

~~if (abs(D$_{ijmn}$) - 2^(ceil(lg(abs(D$_{ijmn}$)+1))-1)-1)) <= abs(D$_{ijmn}$-Q$_{mn}$*S$_{ij}$ * round(D$_{ijmn}$/(Q$_{mn}$*S$_{ij}$)))~~

~~then~~

~~T$_{ij}$ = sign(D$_{ijmn}$)*(2^(ceil(lg(abs(D$_{ijmn}$)+1))-1)-1)~~

if Ernd$_i$[m,n] <= Evq$_i$[m,n] then

T$_i$[m,n] = sign(T$_i$[m,n]) P(abs(T$_i$[m,n]))

```
                }
            }
        }
}
```

The above pseudo-code has the effect of zeroing any coefficients that would have been zeroed if ~~D_{ijmn}~~ Y_i were quantized with the local quantization table S_i ~~Q~~ Q_{mn} ~~$*S_{ij}$~~, but were not zeroed when quantized with the global quantization table S_{min} ~~$QQ_{mn}*S_{min}$~~. Also, it rounds down in magnitude (by one power of two) any coefficient that may be so rounded and not introduce more relative error in reconstruction than if that coefficient were truly quantized by S_i ~~Q~~ Q_{mn} ~~$*S_{ij}$~~. This has the net effect of pseudo-quantizing ~~D_{ij}~~ Y_i with the local table S_i ~~Q~~ ~~$Q*S_{ij}$~~ , while actually quantizing the coefficients with the global table S_{min} ~~Q~~ ~~$Q*S_{min}$~~.

Finally, the quantized blocks ~~T_{ijmn}~~ T_i and the global quantization table S_{min} ~~Q~~ ~~$Q*S_{min}$~~ are entropy encoded at step **58** to create a JPEG Part 1 image file **60** in accordance with the JPEG algorithm while still producing a fully compliant JFIF stream.

It should be noted that the algorithm is particularly useful in optimizing JPEG images that have already been quantized using the standard JPEG quantization table at a level S_{min}. By definition ~~S_{ij}~~ S_i >= S_{min}, hence the algorithm guarantees that the optimized JPEG will never be larger in size than the original JPEG, and will in almost all instances be smaller. At the same

time the pseudo-variable-quantization ensures that the image quality remains essentially unchanged to the human observer.

Page 21, line 21 to Page 23, line 11

Because of the subsampling, there may be up to four Y channel blocks that correspond to the same region of an image covered by one U and V block. In this case, the visual importance ~~I_{ij}~~—I_i that is used in the U and V channels is simply given as,

max {all corresponding ~~I_{ij}~~—I_i values from the Y channel}.

Referring to Figure **3** the apparatus for JPEG Compression using the above algorithm consists of a frame grabber **80** into which non-overlapping, tiled, 8 x 8 image pixel blocks ~~B_{ij}~~—X_i are stored temporarily. For ~~E~~each block, ~~B_i~~X_i, the digital cosine transform (DCT) is calculated by DCT transformer **82** and the resultant transform coefficients ~~D_{ijmn}~~—Y_i stored in memory **84**. A visual importance calculator **86** calculates values of the visual importance, ~~I_{ij}~~—I_i, for each block ~~B_{ij}~~—X_i. A global quantization calculator **87** calculates elements ~~Q_{ij}~~—Q[m,n] of a global quantization matrix utilizing I_i and Y_i, ~~I_{ij}~~, ~~and B_{ij}~~. A linear scaling factor calculator **89** uses user set values of S_{min} and S_{max} set in blocks **124** and **126**, respectively, and ~~I_{ij}~~—I_i to determine ~~S_{ij}~~—S_i in calculator **128** for quantized blocks ~~T_{ij}~~—T_i.

More particularly, values of the quantization matrix ~~Q_{ij}~~ Q are calculated by first forming the sum of the product of the visual importance ~~I_{ij}~~ I_i and the elements of blocks ~~B_{ij}~~ Y_i in block **88** to form the elements ~~A_{mn}~~ A[m,n] in an 8 x 8 array which are stored in memory **100**. The maximum value "Max ~~A_{mn}~~A[m,n]" in the array is selected by Max ~~A_{mn}~~A[m,n] selector **102**. The elements ~~Q_{mn}~~B[m,n] of the intermediate matrix Q B are calculated as ~~(Max A_{mn})/A_{mn}~~ B[m,n] = (Max A[m,n])/A[m,n] in block **104**. The scaling factor s is determined by calculator **129** to minimize the error between s B and user defined standard quantization matrix Q_{std} set in block **130**. The quantization matrix Q is calculated as Q = s B in calculator **131**.

In block 106, the quotient of ~~D_{ijmn}/(S_{min}*Q_{mn}~~ Y_i[m,n]/(S_{min} Q[m,n]) is rounded to the nearest integer yielding elements ~~T_{ijmn}~~T_i[m,n]. In comparator **108**, the calculated value of ~~T_{ijmn}~~T_i[m,n] is compared with zero and, if greater than zero, in block **110** the quotient Y_i[m,n]/(S_i Q[m,n]) is calculated and then rounded to the nearest integer. If the quotient Y_i[m,n]/(S_i Q[m,n]) equals zero, then T_i[m,n] is set equal to zero at block **112**. If the quotient Y_i[m,n]/(S_i Q[m,n]) is not equal to zero, at block **110**, then the value of the rounded value of the latter quotient is transferred to block **116**. Values calculated in blocks **116** and **118** are compared in calculator **120** and if the

value calculated in block **116** is less than or equal to the value calculated in block **118,** then the value of ~~T_{ijmn}~~$T_i[m,n]$ is set equal to ~~sign(D_{ijmn})*(2^(ceil(lg(absD_{ijmn})+1)-1)-1)~~ sign($T_i[m,n]$) P(abs($T_i[m,n]$)). The blocks of quantized coefficients ~~T_{ij}~~ T_i and the global quantization table ~~Q*S~~$_{min}$ S_{min} Q are entropy encoded by entropy encoder **113.**